

Maritime Search and Rescue via Multiple Coordinated UAS

Dieter W. Schuldt · Joel Kurucar

Received: date / Accepted: date

Abstract The advent of Unmanned Aerial Systems (UAS) has created opportunities for expensive capital assets to be replaced by these small, yet capable, platforms. Tasks that are identified as able to be performed by UAS may also benefit from the ability of a collection of UAS to operate in a cooperative and parallel manner. Parallelization also means that several parts of the search area can be covered at the same time, reducing the overall task completion time. In this paper we investigate how to divide the maritime search and rescue task in a way that it can be performed by a set of UAS. Our investigation covers the detection of multiple mobile objects by a heterogeneous collection of UAS. Three methods (two that are not informed by object location probabilities and one that is) for dividing the space are proposed, and their relative strengths and weaknesses investigated. To facilitate simulation a model for object detection by an aerial camera is proposed. The topic is approached holistically to account for contingencies such as airspace deconfliction. Results are produced using simulation to verify the capability of the proposed method and to compare the various partitioning methods. Results from this simulation show that great gains in search efficiency can be made when the search space is partitioned using a method based on object location probability. For search areas of 3km x 3km detection probability gains of 40 percent are achievable. For larger areas these gains can approach 80 percent.

Keywords Unmanned Aerial Systems · Search and Rescue · Cooperative Search · Parallelized Search · Swarm Robotics · Remote Sensing

Dieter Schuldt
MIT Lincoln Laboratory
E-mail: dieter.schuldt@ll.mit.edu

Joel Kurucar
MIT Lincoln Laboratory

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited. This material is based upon work supported by the Department of Homeland Security under Air Force Contract No. FA8721-05-C-0002 and/or FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of Homeland Security.

1 Introduction

This paper is motivated by the idea that using modern technology to adapt and augment established methodologies can improve the results of those tried-and-true techniques. In this paper, the existing paradigm in question is Search and Rescue and the emerging technology is small Unmanned Aerial Systems.

Search and Rescue (SAR), the process of locating and provisioning aid to persons who are, or are feared to be, in need of assistance[25] has a long history and is the subject of well formulated international agreements and organizations[18]. Dating back as early as 1656 A.D., governments and organizations have been documenting SAR efforts as a way of accounting for and learning from these missions[17]. In the United States alone, the United States Coast Guard (USCG) has saved more than 1,500,000 lives since 1790[22]. Annually, since 2000, the USCG has responded to over 47,000 SAR cases and saved over 4,400 lives[22]. While impressive, not all SAR cases end in success. Since 2000, the USCG reports that 7,158 lives were left unaccounted for[22]. These numbers only include maritime SAR events in the United States. If one considers the rest of the world and additional types of SAR (Urban, Land, Wilderness), the impact of research focused on improving SAR is significant.

This paper investigates SAR improvements based on research into the application of small Unmanned Aerial Systems (UAS), specifically readily available consumer UAS, into SAR operations. Based on the near ubiquitous use of small UAS in today's society it is only a matter of time before their usage is wide-spread in the public safety and SAR communities[11].

The specific case used to focus research is maritime SAR. In addition to simplifying concepts such as terrain, this also allows the use of standard and well practiced search patterns. Specifically, this paper narrows the available search patterns to those that may be appropriate in both maritime and land environments.

The growing interest in small UAS research has produced a wide array of literature on the topic, including research directed at SAR efforts. [30] provides a summary of how Unmanned Aerial Vehicles (UAVs), including small UAS, have been applied to maritime SAR, breaking up research into offshore and nearshore SAR operations. In addition, this research highlights the different types and capabilities of UAVs for consideration[30]. For the remainder of this paper, the terms UAV and UAS will be used interchangeably even though there is a slight difference in their formal definition[12]. In addition, this research is currently focused on small UAS (under 55 pounds) that allow vertical take off and landing. It should also be noted at this point that the research primarily focuses on the "search" aspect of SAR as small UAS are not typically capable for achieving a reliable "rescue" effort.

Through this lens, a better classification of literature on the subject for this research should be based on the proposed operations of the small UAS. This paper breaks past research down into three main areas. The first is using UAS to perform single asset search of an area. The second is using UAS in a parallel search of an area. The third is using UAS cooperatively to search an area. The research presented in this paper falls on the border of the second and third approach.

The first area of related research focuses on the use of a single SAR asset. At a foundational level, this paper extends the concepts developed in this research. The primary source of research and doctrine surrounding single asset SAR is provided by government agencies who are charged with performing SAR daily. [25] provides the technical background and justification from a United States perspective and [1] does the same for Canada. In addition, [16] provides research into how to execute SAR in wilderness environments. These research areas provided the underpinnings of this paper.

Parallel search techniques, where several search agents operate within the same area, have also been explored extensively. Examples of methods for operational use of multiple assets to search an area in parallel can be found in [25]. Formation flying, as seen in [14] and [20], yield similar results. While this research is interesting and formative in terms of utilizing multiple assets independently, the search methodology described in this paper is inherently different in nature.

The third area of research focuses on UAS cooperatively searching an area. This paper most closely fits into this set of research although there is one primary key differentiator. The myriad research areas surrounding this topic, [3], [21], [9], [6], [28], [15], and [19] are just a notable few, all depend on some level of communication between UAS while in flight. This paper investigates an approach to teaming and cooperation between assets, however all coordination is done a priori. This means that UAS do not require any communication between each other during the search. While this does limit the potential effectiveness of the result, since dynamic cooperation between assets should only increase effectiveness, this paper shows that the approach described herein is worth pursuing as well, especially when there are significant costs or constraints associated with inter-agent communication. [26] provides an interesting and comprehensive investigation about the costs and benefits associated with agent communication.

The most relevant previous work [29] uses a similar approach to solving the problem of searching using multiple UAS in parallel. However, there are some notable difference in research approaches. The first and most obvious is the nature of the search area partitioning. In [29] partitions are defined differently and are dynamic in nature. In addition, the partitions in this paper are rectangular to accommodate well defined search patterns while in [29] partitions take on more diverse shapes. Finally, the methods used to generate the partitions are significantly different. In this research, the underlying probability model evolves with time, allowing the modeling of search object movement; this is not the case in [29]. Finally, this paper utilizes well defined and widely accepted search patterns taken from real world SAR operations. While this paper does not include the full range of available search patterns, the search patterns included are some of the most widely used in maritime environments[25]. This paper will focus on the key metrics discussed in SAR literature, such as [25], to measure success.

Limiting the search patterns to those that were selected was done for two reasons: to encourage the adoption of this method by SAR workers, and to make the optimization of the path selection more efficient. Using search patterns that are familiar to practitioners in the art of SAR gives them a better understanding of the implications of the flight plans that are created and allows those professionals to easily swap in their own search pattern if they feel that the algorithm is somehow in error. Establishing this trust between the worker and the tool is an important part of human-computer interaction [13].

Other informative sources include research into general multiagent systems, such as those discussed in artificial intelligence research[24]. There is also comprehensive research on related work involving collision avoidance, see [2], [7], [10], and [8], UAS swarms, see [27], [5], [4], and [23], and other interesting UAS developments, all of which tangentially influence and balance this research. These topics are not included in detail since this research is not focusing on collision avoidance systems, air space regulation and policy, or emerging techniques like sense and avoid. Similarly, this research assumes that the UAS cannot communicate with each other, prohibiting the inclusion of swarm techniques.

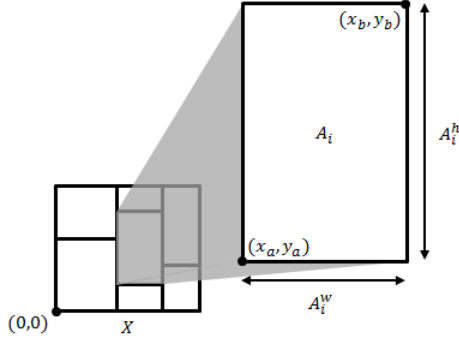


Fig. 1 Complete overview of the search area definitions

2 Definitions

In this problem there are search objects (what is being searched for) and agents (what is doing the searching). The set of agents is $N = \{n_1, \dots, n_u\}$ and the set of objects being searched for is $Q = \{q_1, \dots, q_v\}$. Define the operator that returns the size of a set as $|\cdot|$. Then, the number of UAS is $|N|$ and the number of objects being searched for is $|Q|$.

The area of search X is assumed to be square and is divided into a discrete grid. An element of this grid (called a cell) is designated $\mathbf{x} = (x, y)$. The total width and height of the search area is given by X^w and X^h . The probability of a given object q being located at \mathbf{x} is $P_q(\mathbf{x})$. When talking about the probability of any object being located at \mathbf{x} we will use $P_*(\mathbf{x}) = \sum_q P_q(\mathbf{x}) / |Q|$. We will also treat time as discretized; the search begins at time t_0 and consists of an ordered series of times (t_1, t_2, \dots) thereafter.

A partition set $A = \{A_1, \dots, A_n\}$ of a space X is a disjoint cover of X where $A_i \neq \emptyset \forall i$. For this paper we are interested in rectangular partitions which we will define as $[x_a, x_b] \times [y_a, y_b]$ for two points $(x_a, y_a), (x_b, y_b) \in X$ with $x_a < x_b$ and $y_a < y_b$. Similar to X these values are used to define the width and height of a partition as $A_i^w = x_b - x_a$ and $A_i^h = y_b - y_a$, respectively. This is depicted in figure 1

We will define a UAS and its assignment as a tuple $n_i = \{w, \tilde{A}_i, \gamma, R, \sigma, \theta, (r, s)\}$. Here w is a weight assigning the relative capability of the agent, \tilde{A}_i is the partition to which it is assigned, γ is the trajectory that it will take, R is the range of the agent, σ is a function and θ is a parameter describing the camera being used, and (r, s) is the coordinate location of the UAS. When referring to a property of agent i we will use the designator of the property with the subscript i . For example R_i would refer to the range of agent i . To find the location of an agent at time t we will treat γ as a function $\gamma(t) = \mathbf{x} \in X$.

The search paths are limited to a subset of those defined by the USCG in [25]. We will be using the parallel line, creeping line, spiral, and sector search patterns as seen in figure 2

The distribution of probabilities for objects over an X can be seen in figure 3. Each of the disks is the normal distribution that describes the probabilistic location of the object.

3 Methods

There are several aspects of the search and rescue via multiple UAS problem that need to be addressed for a system to be functional. The following method starts with an area covered

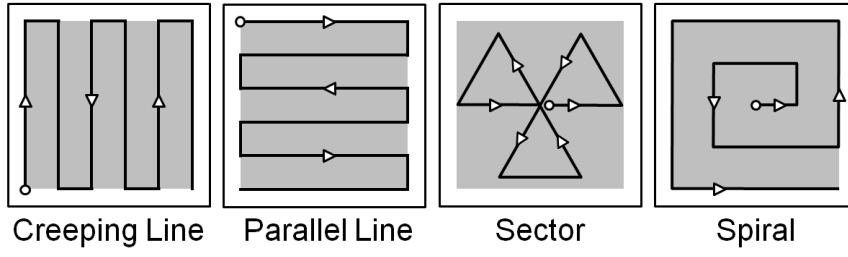


Fig. 2 The four search patterns that are being considered in this paper

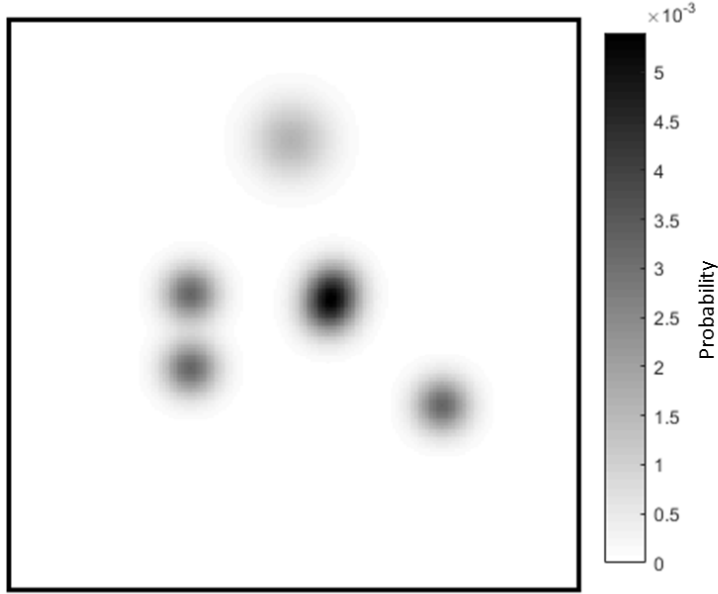


Fig. 3 An example of a search area with 5 search objects represented probabilistically

by some probability distribution and produces a covering set of partitions. Additionally, it assigns each member of a collection of agents a search path, a unique partition, and ensures that minimum separation is maintained throughout the search while minimally impacting optimality.

3.1 Setup

Based on some prior knowledge an initial location probability distribution is assigned to each object. For the purposes of this paper the distribution is defined as a 2 dimensional normal distribution centered at some $\mu \in X$ with covariance matrix $C = c\mathbf{I}$. This can be expressed as

$$\mathcal{N}(\mu, c) = \frac{1}{\sqrt{(2\pi)^2 C}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu)^T C^{-1}(\mathbf{x} - \mu)\right) \quad (1)$$

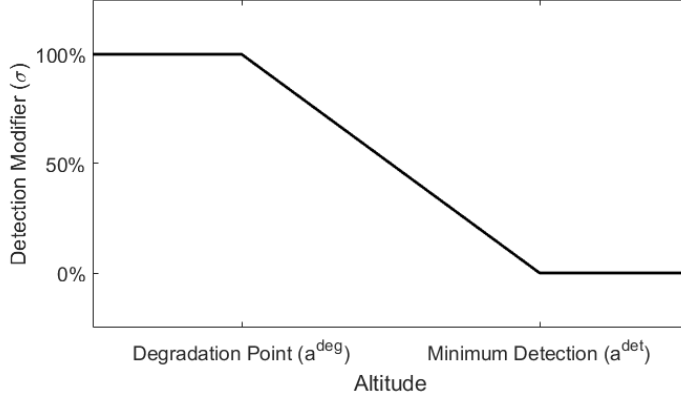


Fig. 4 Vision model used to describe effect of altitude on detection probability

The probability of an object being located at \mathbf{x} can then be found by evaluating this distribution at that point. Since X is discrete the probability assigned to a given discrete grid element is the probability from 1 evaluated at the center of the element. The normal distribution is not compactly supported so for simulation purposes the distribution was extended to square area centered at μ with width equaled to $6\sqrt{c}$. The distribution was also normalized so that it summed to 1 over its support.

In a similar manner we define a motion model M as a 2 dimensional normal distribution (see: 1) with variance Ω . This approximates a random walk by the object. Convolving the probability distribution with the motion model gives the updated distribution at a future time.

$$P_q^{i+1}(X) = M \circ P_q^i(X). \quad (2)$$

3.2 Vision Model

A model was developed to describe how detection probability is affected by altitude for a standard electro-optical system. Given a ground resolution requirement there is some altitude above which the sensor cannot resolve objects of the specified size. In the model this is labeled as a^{det} , the 'minimum detection' altitude. On the other end is a^{deg} , the "degradation point"; a minimum altitude below which the probability of detection is not improved. In between we modeled the degradation as linear. This model is applied as a coefficient affecting the probability of detection $\hat{P}_*(\cdot) = \sigma(a) \cdot P_*(\cdot)$. A plot of $\sigma(a)$ can be seen in figure 4.

The vision model also specifies how much of the search area is visible to the UAS at a given time. Using θ (the camera view angle) a simple trigonometric relationship gives the visible area $V \in X$. With the UAS located at (s, t) at the altitude a the area visible is thus

$$V = [s - a \cdot \cos(\theta/2), s + a \cdot \cos(\theta/2)] \times [t - a \cdot \cos(\theta/2), t + a \cdot \cos(\theta/2)] \quad (3)$$

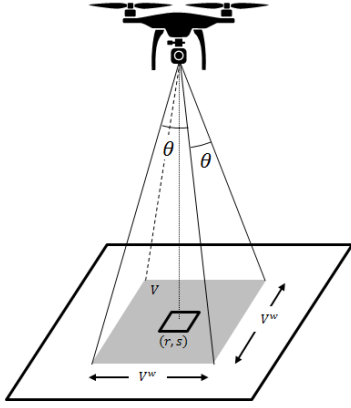


Fig. 5 Visible area diagram

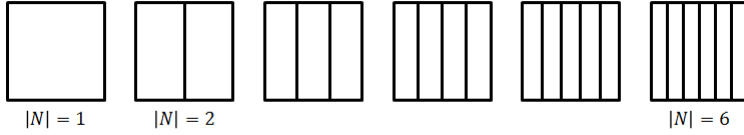


Fig. 6 Track partitioning of a space with 1 through 6 UAS

3.3 Partitioning

3.3.1 Track Partitioning

The most basic partitioning method is to divide the search area into "tracks". This method does not take into consideration the underlying data or attempt in any way to optimize the shape of the partitions. In this method we use

$$A = \left\{ [(i-1) \cdot X^w / |N|, i \cdot X^w / |N|] \times [0, X^h] \mid i = 1, \dots, |N| \right\} \quad (4)$$

3.3.2 Block Partitioning

A slightly more sophisticated method is the following block partition which attempts to produce partitions as efficient as possible without using the underlying data. Partitions are created in such a manner that they all have the same area and their aspect ratios (equation 5) are minimized.

$$AR(A_i) = \max(A_i^w, A_i^h) / \min(A_i^w, A_i^h) \quad (5)$$

In addition to having balanced search areas (as in the track method) this establishes a much smaller mean maximum distance to an edge as can be seen in figure 7. The smaller distances mean that starting on an arbitrary edge is less costly.

The first ten block divisions are depicted in figure 8.

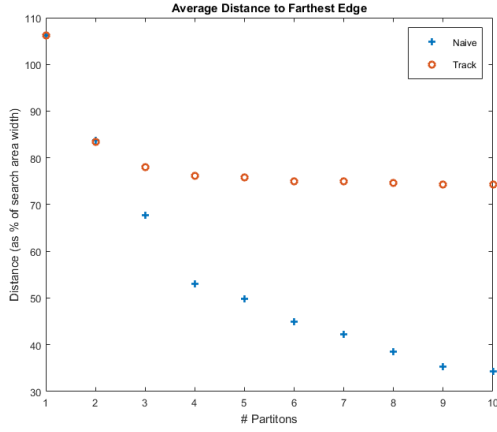


Fig. 7 Average distance in a partition to the farthest edge for Track and Block partitioning schemes

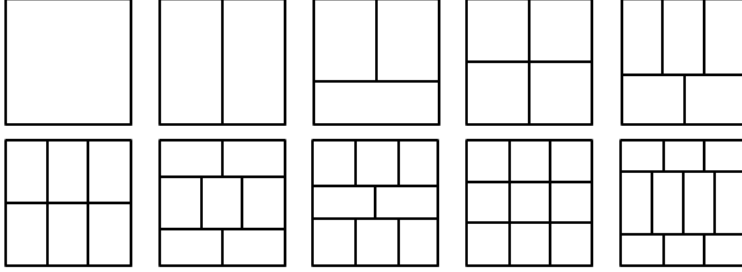


Fig. 8 Block partitioning of a space with 1 through 10 UAS

3.3.3 Informed Partitioning

The final partitioning method uses the underlying probability distribution assumptions to place that probability near the geometric center of the partitions. There are two primary equations used in the proposed partitioning scheme: center of mass, and dispersion. The center of mass is the location of the average probability and the dispersion measures how spread out the distribution is.

The center of mass of $A_i = [x_a, y_a] \times [x_b, y_b]$ is defined as

$$\mathbf{R}(A_i) = \frac{1}{P_*(A_i)} \left(\sum_{u=x_a}^{x_b} \sum_{v=y_a}^{y_b} u \cdot P_*((x_u, y_v)), \sum_{v=y_a}^{y_b} \sum_{u=x_a}^{x_b} v \cdot P_*((x_u, y_v)) \right) = (R_x, R_y) \quad (6)$$

Dispersion is the geometric variance and is defined as

$$D(A_i) = \frac{1}{P_*(A_i)} \sum_{u=x_a}^{x_b} \sum_{v=y_a}^{y_b} \left(\left(v - \frac{x_a + x_b}{2} \right)^2 + \left(u - \frac{y_a + y_b}{2} \right)^2 \right) \cdot P_*((x_u, y_v)) \quad (7)$$

The objective is to maximize the detection probability by partitioning in such a way that the distribution in a partition is spread out in a manner conducive to searching. Because

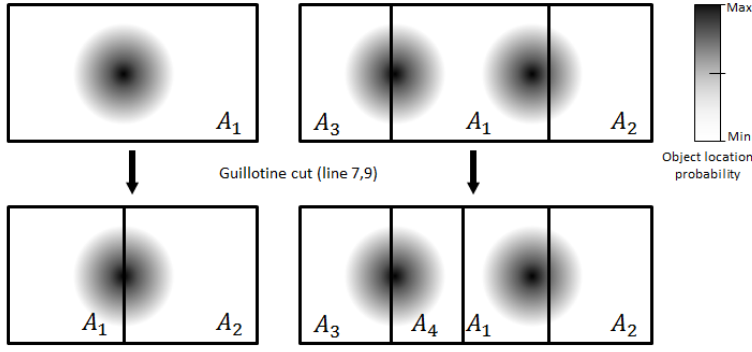


Fig. 9 (left) The case where a partition has a large amount of probability assigned to cells in it. After a cut is made the two resulting partitions have approximately even total probabilities. (right) A_1 has a large amount of probability far away from its geometric center. Dividing this partition results in two partitions A_1 and A_4 with approximately equal probability, and the probability is more centrally located in each.

of their limited range you cannot guarantee that an agent will be able to cover the entire partition. This means that it is beneficial to have the vast majority of the probability in a partition located either at the geometric center or along a single edge. The following algorithm partitions the space advantageously, as depicted in 9.

This algorithm recursively divides the search space using guillotine cuts, so for convenience we will define the operation $A_i \leftrightarrow (A_j, A_k)$ to mean "replace the partition A_i in A with the partitions A_j and A_k ". We will also define the operation $(A_j, A_k) = \text{HORIZ}(A_i, y)$ as the separation of A_i along a horizontal line drawn at y . That is, given $A_i = [x_a, x_b] \times [y_a, y_b]$ we have $\text{HORIZ}(A_i, y_c) = ([x_a, x_b] \times [y_a, y_c], [x_a, x_b] \times [y_c + 1, y_b])$. The operation $\text{VERT}(A_i, x_c)$ is defined in a similar manner but with the division running vertically.

Algorithm 1 Partitioning Algorithm

```

1: while  $|A| < |N|$  do
2:    $div \leftarrow i \text{ s.t. } D(A_i) > D(A_j | j \neq i)$ 
3:    $(com_x, com_y) = \mathbf{R}(A_{div})$ 
4:    $(H_1, H_2) = \text{HORIZ}(A_{div}, com_y)$ 
5:    $(V_1, V_2) = \text{VERT}(A_{div}, com_x)$ 
6:   if  $AR(H_1) * AR(H_2) > AR(V_1) * AR(V_2)$  then
7:      $A_{div} \leftarrow \text{HORIZ}(A_{div}, com_y)$ 
8:   else
9:      $A_{div} \leftarrow \text{VERT}(A_{div}, com_x)$ 
10:  end if
11: end while

```

On line 2 of Algorithm 1 selecting the partition with the maximum dispersion results in a new set of partitions that is more balanced and efficiently searchable. There are two conditions that result in a partition having a large dispersion: a large total probability or a probability distribution that is predominantly far from the geometric center. In the first case, dividing the partition results in two partitions with probabilities more balanced with the other partitions in the set. In the latter the division will produce partitions with lower dispersion, meaning more concentrated probability.

Taking into account the aspect ratio (line 6) works to create partitions that are more square. Having square partitions increase the probability of detection by encouraging the use of sector searches and by minimizing the distance to furthest edge. The latter is particularly important in searches of large areas where the range of the UAS is insufficient to cover the entire partition it is assigned to.

3.4 UAS Assignment

After search space partitioning, the next step is to assign each agent to a unique partition. The task allocation algorithm chosen for this research is a simple greedy approach aimed at pairing the most capable UAS with the most important partition. We refer to the most capable UAS as the one with the highest quality camera and longest endurance. This concept is supported by [16], which highlights the criticality of camera quality in search events. The weighting assigned as w_i is a weighted combination of camera quality and endurance. We will use a_i^{det} as the measure of camera quality and R_i as the endurance.

$$w_i = 0.8 \cdot \left(\frac{\max(a_j^{det}) - a_i^{det}}{\max(a_j^{det}) - \min(a_j^{det})} \right) + 0.2 \cdot \left(\frac{\max(R_j) - R_i}{\max(R_j) - \min(R_j)} \right) \quad (8)$$

Similarly, the most important partition is the partition with the largest ratio of search object probability per area. We will define this density as

$$A_i^D = \frac{P_*(A_i)}{A_i^w \cdot A_i^h} \quad (9)$$

By a slight abuse of notation we will also consider the array A^D as the ordered set $\{A_i^D\}$.

The task assignment algorithm is defined in Algorithm 2. It is important to note that this algorithm assumes the number of partitions and the number of UAS are equal. Without this assumption, the task assignment algorithm presented is no longer appropriate.

Algorithm 2 Task Allocation Algorithm

```

1: for each  $n_i$  do
2:   Calculate  $w_i$ 
3: end for
4: for each  $A_i$  do
5:   Calculate  $A_i^D$ 
6: end for
7: Sort  $N$  by  $w_i$ 
8: Sort  $A^D$  by  $A_i^D$ 
9: for  $i = 1 : |N|$  do
10:   $\tilde{A}_i \leftarrow A_i^D$ 
11: end for

```

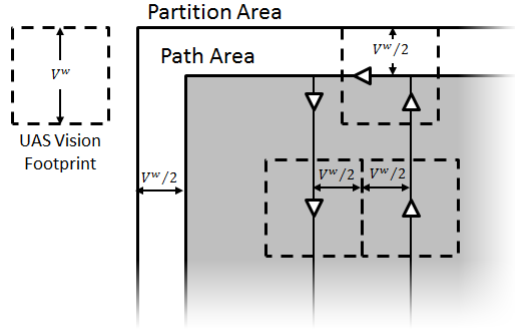


Fig. 10 Path coverage area and method of path scaling

3.5 Path Evaluation

Probability is collected using a push broom method. At each moment in discrete time there is some area beneath the UAS that is being viewed by the camera of the UAS. There is some probability of the objects being searched for being in that area. All of this probability is summed up to acquire the overall probability of detection of both any object and each object.

During the path planning stage an objective function is maximized to give the best path and altitude for each partition.

$$\arg \max_{\gamma, a} \sigma(a) \cdot \tilde{P}_*(\gamma) \quad (10)$$

As specified in the Definitions section there are four possible paths that a UAS can be assigned. Each of these paths is scaled to fill the Path Area of the partition. This is detailed in figure 10.

The starting location of patterns can be moved by mirrored the pattern along the horizontal or vertical axis (or both), giving 16 total paths that have to be evaluated. Constraining the problem to this set of paths allows the optimization to be performed in a reasonable time. Assuming that the object is moving much slower than the UAS additional savings can be made by evaluating $P_*(\gamma)$ only some of the points of γ .

Given γ , the path traced by the UAS, we will consider the subsequence γ_t for $t = t_1, t_2, \dots, t_m$, with $t_1 < t_2 < \dots < t_m$. At each of these points the area visible to the UAS is denoted V^t .

Through the camera model we will adjust the posterior distribution of the object locations after each observation is made

$$P^{\tau_{k+1}}(V) = (1 - \sigma(a)) \cdot P^{\tau_k}(V) \quad (11)$$

The probability of a given UAS detecting a single object q is the sum of that object's probability accumulated by an agent during its flight given by

$$\tilde{P}_q = \sum_{k=1}^m P_q^{\tau_k}(V^{t_k}) \quad (12)$$

and the total overall probability of detecting any object is

$$\tilde{P}_* = 1 - \prod_{i=1}^{|N|} \left(1 - \sum_{k=1}^m P_i^{t_k}(V^{t_k}) \right) \quad (13)$$

This same procedure is attempted for a range of potential altitudes to determine what altitude produces the maximum detection probability.

3.6 Deconfliction

To understand the deconfliction algorithm we will establish a few terms. Consider the set of desired altitudes specified in the UAS Assignment section. Let the sequence J be defined as the list of unique IDs in the priority order. Then, $J_i = 1, \dots, |N|$ produces a map such that a_{J_1} is the altitude of the UAS with the highest priority, and so forth through $a_{J_{|N|}}$ the altitude of the UAS with the lowest priority. Similarly, allow J to recover the UAS (meaning n_{J_1} is the tuple of the highest priority UAS). We want to assign altitudes to each UAS such that $d(a_i - a_{i+1}) \geq a_{min}$ which results in a small disruption to the desired altitudes. The value a_{min} is the minimum separation and is used to account for instrument error and process noise due to wind, etc. For this paper the assigned priorities were sequential integers in inverse weighting order (e.g. given 5 UAS the most important would be assigned weight 5, the second most important 4, etc.)

In assigning the altitudes we will start from the highest priority UAS and progressively assign each UAS a new altitude near the desired altitude. While assigning altitudes, we will use the function $\hat{a} = A(a)$ to retrieve the altitude of the UAS assigned an altitude nearest and above a and $\check{a} = B(a)$ to retrieve the altitude of the UAS assigned an altitude nearest and below a . If there are no assigned altitudes that meet these criteria then ∞ and 0 are returned, respectively.

We will also make use of a normalizing algorithm that calculates the total *cost* of rearranging to meet minimum spacing. This algorithm can be used in two modes: UPDATE and CHECK. In UPDATE mode the altitudes are adjusted, whereas in CHECK a score is calculated but no altitudes are reassigned.

Algorithm 3 Normalize

```

1: Input  $J$  ▷ The number of agents that have already been assigned
2:  $cost = 0$ 
3: for  $i = 1:J$  do
4:   if  $d(a_{j_i}, a_{j_{i+1}}) < a_{min}$  then
5:      $cost += w_{j_i} \cdot d(a_{j_i}, a_{j_{i+1}})$ 
6:     if UPDATE mode then
7:        $a_{j_{i+1}} = a_{j_i} + a_{min}$ 
8:     end if
9:   end if
10: end for
11: Return  $cost$ 

```

With these preliminaries out of the way we can now show how to assign altitudes to the UAS.

The end result is all of the agents being assigned to their own vertically segregated airspace with a minimum separation of a_{min} . This means that as long as the agents start

Algorithm 4 Deconflict

```

1: for  $i = 1 : |N|$  do
2:   if  $A(a_{j_i}) - a_{j_i} > a_{min} \ \& \ a_{j_i} - B(a_{j_i}) > a_{min}$  then
3:      $a_{j_i} = a_{j_i}$  ▷ No adjustment is necessary
4:   else if  $A(a_{j_i}) - B(a_{j_i}) > 2 \cdot a_{min}$  then
5:     if  $A(a_{j_i}) - a_{j_i} < a_{j_i} - B(a_{j_i})$  then
6:        $a_{j_i} = A(a_{j_i}) - a_{min}$ 
7:     else
8:        $a_{j_i} = B(a_{j_i}) + a_{min}$ 
9:     end if
10:  else
11:     $best = 0$ 
12:     $minScore = \infty$ 
13:    for  $k = 1 : i$  do
14:       $a_{j_i} = B(a_{j_k})$ 
15:      if  $minScore > Normalize(i)$  then
16:         $minScore = w(a_{j_i}) \cdot (B(a_{j_k}) + a_{min} - a_{j_i}) + Normalize(i)$ 
17:         $best = k$ 
18:      end if
19:    end for
20:     $a_{j_i} = B(a_{j_{best}})$ 
21:  end if
22:   $Normalize(i).UPDATE$ 
23: end for

```

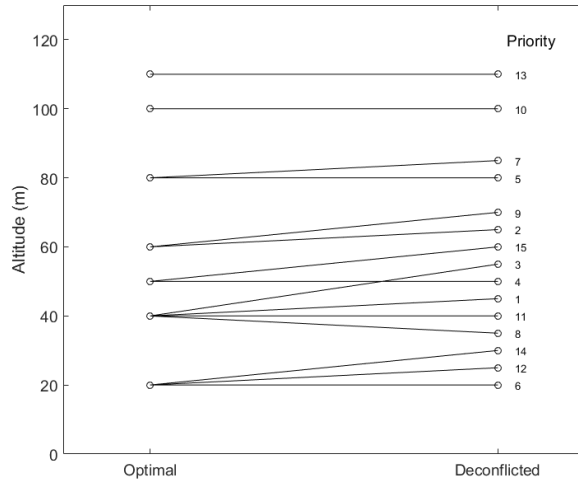


Fig. 11 Depiction of initial altitudes as produced by path evaluation optimization and the final altitudes after application of the deconfliction algorithm

outside of X and ascend vertically to their designated altitude they will always be separated by at least a_{min} . A visualization of the outcome of this procedure is provided in figure 11.

4 Results

To determine the performance of these methods and models a Monte Carlo simulation was performed. Each instance of this simulation took as inputs the size of the search area and the number of objects that were being searched for. This allowed us to evaluate the effectiveness of each partitioning method for a range of scenarios. Each set of parameters was evaluated 100 times with a different random seed each time.

The grid on X was set such that each square represented an area of 1 m^2 . Each object was given a uniformly distributed random variance $c \sim \mathcal{U}(5, 10) \text{ m}$. Object positions were randomly assigned to $[3\sqrt{c}, X^w - 3\sqrt{c}] \times [3\sqrt{c}, X^w - 3\sqrt{c}] \subset X$. The motion model M was given covariance 1 m .

Each UAS was given a range (R_i) of $8400 + \mathcal{U}(-500, 500) \text{ m}$. Altitude restrictions were imposed to mandate an altitude between 12 m and 120 m. The point of Degradation was set at 25 m and the Minimum Detectable altitude at 300 m. The minimum separation was set at $a_{min} = 5 \text{ m}$. Based on standard available consumer cameras θ_i was set at 90 degrees.

Define the average probability of detection as

$$P_{mean} = \frac{\tilde{P}_*}{|Q|} \quad (14)$$

Simulations were performed for square X with lengths of 1500, 3000, 4500, and 6000 m and with between 1 and 4 objects. The P_{mean} distributions of these simulations can be seen in figures 12 - 15. Altitudes for the naive methods were selected to be optimal even though this would be impossible in practice given the assumption in those cases that the underlying probability is not available. A box plot is used to display the results from the informed partitioning method while the two naive methods are plotted with markers at their mean and lines at ± 1 standard deviation.

These results can be better visualized as the difference in mean probability of detection between the informed and naive methods. In this case the mean value of P_{mean} was used and (owing to their similarity) the two naive methods were averaged together.

This data is used to investigate the impact of changes in search area size. In figure 17 the plot shows the difference between the minimum and maximum probability of detection for a given number of UAS ($P_{mean}^{informed} - P_{mean}^{naive}$) for each of the search areas.

Since the curves do not vary significantly over the number of search objects present (less than 10 percent in most cases) we can average over that parameter to arrive at a reduced space that is more easily manipulated. In 16 we've used this reduced set to show the difference in detection probability between the informed and naive methods.

The next test results come from partitioning based on partial knowledge of the object locations. Simulations were generated with 4 search objects, but partitioned based on only some subset of those objects. This represents a case where a report is made of some number of objects, but the observer does not know how many objects there might be.

5 Discussion

As can be seen from the results applying knowledge about the probable location of the objects being searched for can result in a marked increase in detection likelihood. This extra information allows for the best path, altitude and partition to be selected resulting in much more efficient parallelization of the search. Depending on the size of the search area and the

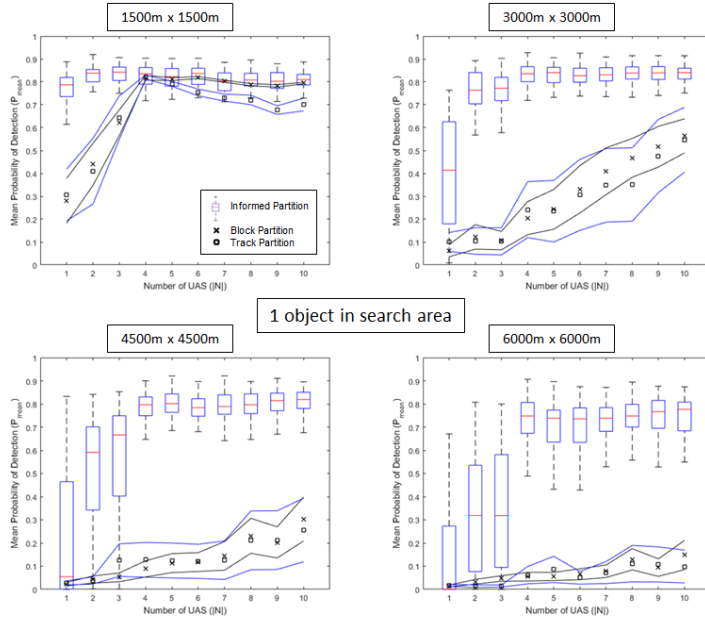


Fig. 12 Mean probability of object detection (P_{mean}) for search areas of (L to R, T to B) 1500, 3000, 4500, and 6000 m with 1 search object

number of objects an increase in mean probability of detection of up to 75 percent can be accomplished, with a mean increase of 47 percent.

Increasing search areas drastically affect the mean probability of detection when using the naive partition methods, whereas the effects on the informed method is generally smaller. Even if we take out the 1500m case where the probabilities are approaching the maximum possible we still see that increasing area has a bigger impact on naive methods for large numbers of UAVs. This further emphasizes the improvements to parallelization that can be achieved through the informed method.

An analytic solution to selecting the most efficient altitude would speed up computation but is not as straight forward to develop as it initially appears. While the detection coefficient σ goes down linearly and the area of observation increases quadratically this is not enough to determine the altitude that accumulates the highest probability. Since the total area that can be covered in the allotted time is also affected by the altitude that must be accounted for as well as the layout of the probability within the partition. A concise analysis may provide a good rule, but the complications involved in this complex and dynamic system serve to confound attempts at some closed form solution.

It is interesting to note that the detection probability is not monotonic for high $|N|$ over small X . This is caused by probability leaking across partitions into areas that have already been covered by the UAS assigned to that partition. For small $|N|$ the average distance to an edge is sufficiently large to ensure that most of the probability stays in the partition that it starts in, however as the partitions get smaller this distance also decreases. The two naive partitioning methods strikingly show off this phenomenon as the track partitions can very easily leak probability across partitions due to their close proximity. Objects originating in

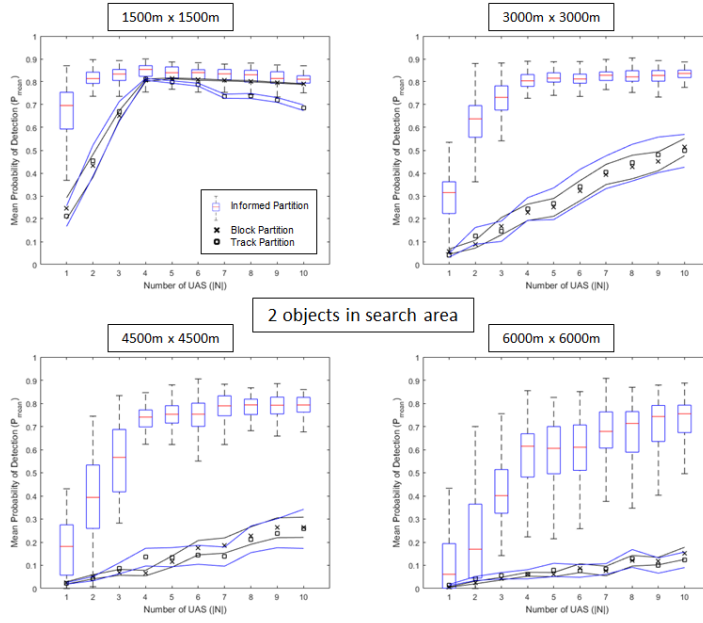


Fig. 13 Mean probability of object detection (P_{mean}) for search areas of (L to R, T to B) 1500, 3000, 4500, and 6000 m with 2 search objects

one partition can end up several partitions away, and it is likely that some number of the UAS that would have detected it had already traveled past the area. For this reason it is recommended that the block partitioning scheme be used in the absence of knowledge about the object locations.

A small anomaly occurs in the informed partitioning at $|N| = 3$, where the mean detection probability is lower than would be expected from the overall trend. This is an artifact of the use of guillotine cuts. The first cut divides the space into two areas of approximately equal total probability, but the second cut operates on only one of these partitions, resulting in a split of 25/25/50 percent between the three partitions. This leaves one UAS to try and do almost as much searching as either of the others. For higher numbers of partitions the expected difference is much lower so the effect is not seen nearly as much in the 5 UAS case where the difference is anticipated to be around 12 percent instead of 25.

For every combination of area and object count there is a significant jump in P_{mean} for the informed case when the number of agents first exceeds the number of objects. This is because having fewer UAS than objects means some number of the agents are "responsible" for detecting multiple objects. These objects could be quite geographically separated so there is no guarantee that the UAS will have sufficient endurance to observe all of the objects that have a high probability of being in their partition. There is also a point above which adding UAS is no longer significantly beneficial. Based on the nominal agent selected for this investigation it takes approximately one agent to search 1 km². As the areas increase, though, there is some savings on that such that 20 km² can be quite reasonably searched by as few as 8 UAS (depending on the number of objects). This reasonable number agents

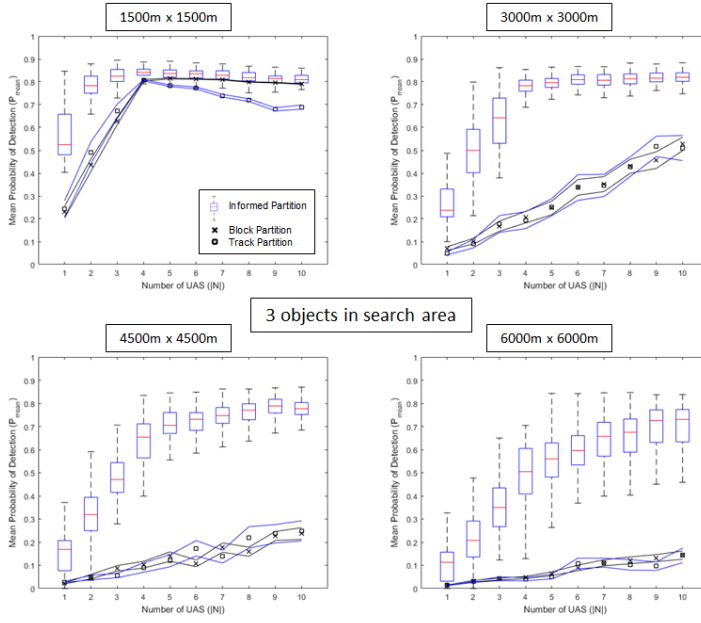


Fig. 14 Mean probability of object detection (P_{mean}) for search areas of (L to R, T to B) 1500, 3000, 4500, and 6000 m with 3 search objects

means that the system could be used even by the cash strapped first response agencies of today.

In the partial information case we see that for small areas (1500 - 3000 m) if we only have knowledge about the location of 1 or 2 of the 4 objects then for some number of agents its is better to use a naive scheme. This is the result of what is essentially overfitting from the partitioning algorithm. During partitioning the known locations are accommodated, but the unaccounted for objects are placed into geometrically unfavorable conditions. The assigned search patterns will only be optimized for the known probabilities, resulting in detection of the unknown objects only through serendipity. This is an important result that needs to be taken into consideration when selecting an algorithm for a given mission.

6 Conclusion

As UAS become more capable and less expensive they are found to be a viable solution to an increasing range of problems. Search and rescue currently relies heavily on capital assets, such as helicopters, that are expensive to operate and maintain. Replacing the current operation methods with one that uses UAS performing in parallel reduces the cost and potentially speeds up the process. A scalable solution also ensures that it is applicable to a wider set of circumstances. Reducing the manned requirement also means that these systems can be operated in more dangerous situations, as their loss is significantly less costly than a manned asset. This work is one step towards creating an overarching framework for UAS SAR operations.

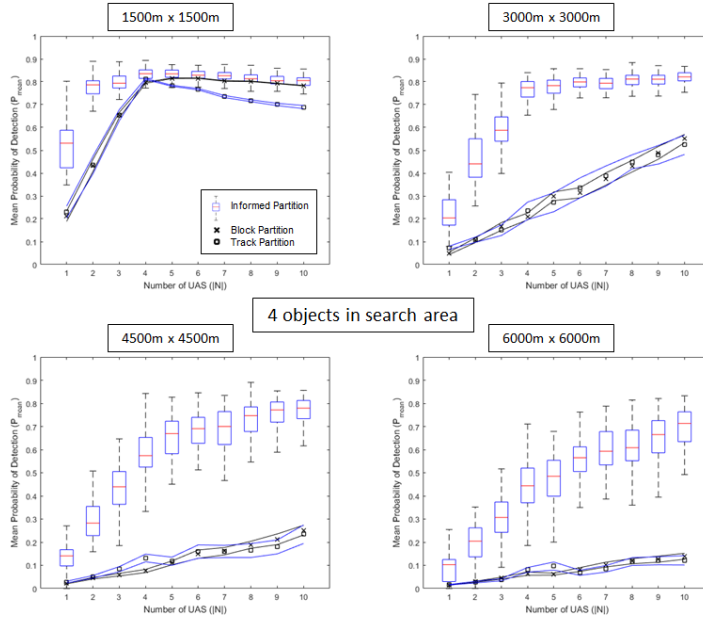


Fig. 15 Mean probability of object detection (P_{mean}) for search areas of (L to R, T to B) 1500, 3000, 4500, and 6000 m with 4 search objects

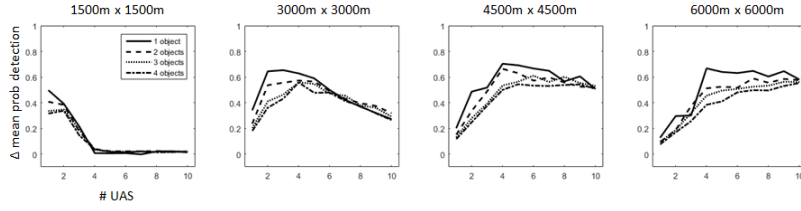


Fig. 16 Difference in Mean probability of object detection (P_{mean}) between the informed and naive methods for search areas of (L to R) 1500, 3000, 4500, and 6000 m

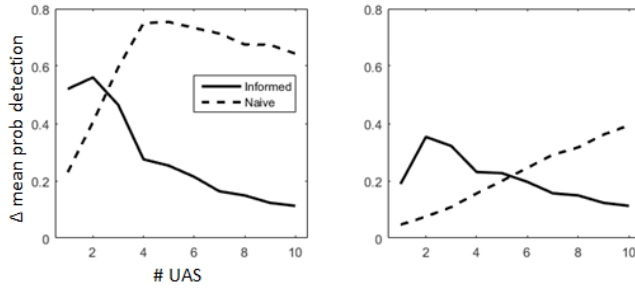


Fig. 17 Change in mean probability of object detection (P_{mean}) (Left) across all areas and (Right) across 3000m to 6000m

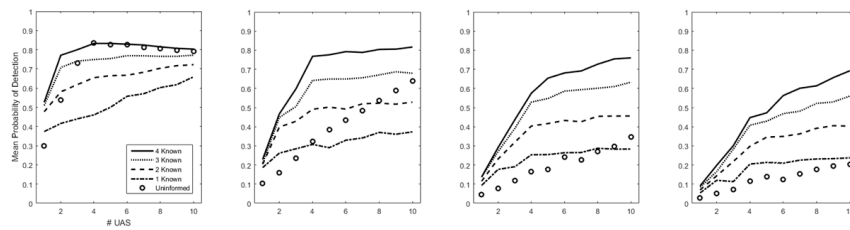


Fig. 18 Mean probability of object detection (P_{mean}) for a variety of knowledge conditions for search areas of (L to R) 1500m, 3000m, 4500m, and 6000m

Laid out in this paper is a method that efficiently divides a space for search by a collection of heterogeneous UAS. The process ensures that the UAS maintain vertical airspace segregation to avoid collisions. It does not make any assumptions about communication between the agents, meaning that there is no barrier to inclusion of a diverse set of agents and allows it to be employed in areas where RF propagation may be limited. This paper also accounts for multiple targets (independent of the number of agents), and adapts to incorporate any information that is available about the targets' locations. A temporal element incorporates potential motion by the targets. This paper also provides a model for how altitude affects the capability of aerial cameras to perform search operations.

Our results show that even in the absence of knowledge about the location of the targets a scheme that attempts to create low aspect ratio partitions performs better than simple track partitioning. Further, taking into consideration the underlying probabilities can increase detection rates even more.

Future work should include refining the vision model and applying modifiers to account for terrain. The restriction to square partitions made in this paper should also be relaxed to explore how other methods (Voronoi for example) perform. The paths restriction was made to appeal to current SAR workers, but as an understanding of this method reaches the larger community it may be possible to explore new search patterns that perform even better.

The authors would like to thank Dr. Paul Breimyer, Dr. Franz Busse, Amna Greaves, and Andrew Weinert

References

1. Abi-Zeida, I., Frost, J.R.: SARPlan: A Decision Support System for Canadian Search and Rescue Operations. *European Journal of Operational Research* **162**(3), 630–653 (2005). DOI 10.1016/j.ejor.2003.10.029
2. Bachrach, A., He, R., Roy, N.: Autonomous Flight in Unknown Indoor Environments. *International Journal of Micro Air Vehicles* **1**, 217–228 (2009). DOI 10.1260/175682909790291492
3. Bethke, B., How, J.P., Vian, J.: Multi-UAV Persistent Surveillance with Communication Constraints and Health Management. *Guidance, Navigation, and Control Conference* pp. 1–23 (2009)
4. Brust, M.R., Strimbu, B.M.: A Networked Swarm Model for UAV Deployment in the Assessment of Forest Environments. *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015 IEEE Tenth International Conference on* pp. 1–6 (2015). DOI 10.1109/ISSNIP.2015.7106967

5. a. Bürkle, Segor, F., Kollmann, M.: Towards Autonomous Micro UAV Swarms. *Journal of intelligent & robotic systems* **61**, 339–353 (2011). DOI 10.1007/s10846-010-9492-x
6. Chai, G., Lin, C., Lin, Z., Zhang, W.: Consensus-Based Cooperative Source Localization of Multi-Agent Systems with Sampled Range Measurements. *UNMANNED SYSTEMS* **2**(3), 229–239 (2014). DOI 10.1142/S2301385014400020
7. Cook, S., Lacher, A., Maroney, D., Zeitlin, A.: UAS Sense and Avoid Development - the Challenges of Technology, Standards, and Certification. 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition pp. 1–10 (2012). DOI 10.2514/6.2012-959
8. Cook, S.P., Lacher, A.R., Maroney, D.R., Zeitlin, A.D.: UAS Sence and Avoid Development - the Challenges of Technology, Standards, and Certification. Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition (2012). DOI 10.2514/6.2012-959
9. Geng, L., Zhang, Y.F., Wang, J., Fuh, J.Y.H., Teot, S.H.: Cooperative Mission Planning with Multiple UAVs in Realistic Environments. *UNMANNED SYSTEMS* **2**(1), 73–86 (2014). DOI 10.1142/S2301385014500058
10. Griffith, J.D., Edwards, M.W., Mirafior, R.M., Weinert, A.: Due Regard Encounter Model Version 1.0 (2013)
11. Gupta, S.G., Ghonge, M.M., Jawandhiya, P.M.: Review of Unmanned Aircraft System (UAS). *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)* **2**, 1646–1658 (2013)
12. FAA Modernization and Reform Act of 2012. Public Law No: 112-95 (2012)
13. John Lee, N.M.: Trust, control strategies and allocation of fuction in human-machine systems **35**(10)
14. Jr, W.H.E., Martin, P., Mangharam, R.: Cooperative Flight Guidance of Autonomous Unmanned Aerial Vehicles. CPS Week Workshop on Networks of Cooperating Objects (CONET) (2011)
15. Khan, U.M.: Optimal Partitioning of a Surveillance Space for Persistent Coverage using Multiple Autonomous Unmanned Aerial Vehicles: An Integer Programming Approach (2014)
16. Lin, R.L.: UAV Intelligent Path Planning for Wilderness Search and Rescue (2009)
17. Major, R.H.: Early Voyages to Terra Australis, now called Australia, 1 edn. Hakluyt Society (1859)
18. National Search and Rescue Committee: United States National Search and Rescue Supplement to the International Aeronautical and Maritime Search and Rescue Manual
19. Rubio, J.C., Vagners, J., Rysdykz, R.: Adaptive Path Planning for Autonomous UAV Oceanic Search Missions. *Intelligent Systems Technical Conference* pp. 1–10 (2004). DOI 10.2514/6.2004-6228
20. Ryan, A., Hedrick, J.K.: A Mode-Switching Path Planner for UAV-Assisted Search and Rescue. *Decision and Control, 2005 and 2005 European Control Conference. 44th IEEE Conference on* pp. 1471–1476 (2005). DOI 10.1109/CDC.2005.1582366
21. Sabo, C., Kingston, D., Cohen, K.: A Formulation and Heuristic Approach to Task Allocation and Routing of UAVs under Limited Communication. *UNMANNED SYSTEMS* **2**(1), 1–17 (2014). DOI 10.1142/S2301385014500010
22. Schaefer, R.: United States Coast Guard Search and Rescue Summary Statistics 1964–2013. Online (2014). URL [http://www.uscg.mil/hq/cg5/cg534/SARfactsInfo/SAR Sum Stats 64-14.pdf](http://www.uscg.mil/hq/cg5/cg534/SARfactsInfo/SAR%20Summary%20Statistics%2064-14.pdf)
23. Smalley, D.: LOCUST: Autonomous, Swarming UAVs Fly into the Future. Online (2015). URL <http://www.onr.navy.mil/Media-Center/Press-Releases/2015/LOCUST->

- low-cost-UAV-swarm-ONR.aspx
24. Sycara, K.P.: Multiagent Systems. *AI Magazine* **19**(2), 79–92 (1998). DOI 10.1609/aimag.v19i2.1370
 25. Thomas, C.B.: U. S. Coast Guard Addendum to the United States National Search and Rescue Supplement (NSS) to the International Aeronautical and Maritime Search and Rescue Manual (IAMSAR). U.S. Coast Guard
 26. Unhelkar, V.V., Shah, J.A.: ConTaCT: Deciding to Communicate during Time–Critical Collaborative Tasks in Unknown, Deterministic Domains. *Artificial Intelligence, Thirtieth AAAI Conference on* (2016)
 27. Wenga, L., Liua, Q., Xiaa, M., Song, Y.: Immune Network-Based Swarm Intelligence and its Application to Unmanned Aerial Vehicle (UAV) Swarm Coordination. *Neuro-computing* **125**, 134–141 (2014). DOI 10.1016/j.neucom.2012.06.053
 28. Wood, J., Hedrick, J.K.: Space Partitioning and Classification for Multi–target Search and Tracking by Heterogeneous Unmanned Aerial System Teams. *American Institute of Aeronautics and Astronautics* pp. 1–11 (2011). DOI 10.2514/6.2011-1443
 29. Wood, J., Hedrick, J.K.: Space Partitioning and Classification for Multi–Target Search and Tracking by Heterogeneous Unmanned Aerial System Teams. *Infotech Aerospace* pp. 1–11 (2011). DOI 10.2514/6.2011-1443
 30. Yeong, S.P., King, L.M., Dol, S.S.: A Review on Marine Search and Rescue Operations Using Unmanned Aerial Vehicles. *International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering (ISSRI)* **9**(2), 396–399 (2015). DOI scholar.waset.org/1999.8/10001953